

REMARKS

The claims remaining in the present application are Claims 1-23.

35 U.S.C. §103

Claims 1-23 stand rejected under 35 U.S.C. §103 as being unpatentable over Mowbray, "The Essential CORBA," pages 35-53, 184-187, 212-213, and 250-255 (hereinafter, Mowbray), in view of UCS, "UCS Architecture" pages 1-2 (hereinafter, UCS). The rejection is respectfully traversed for the reasons below.

Claim 1 recites:

A method for allowing communication between a Practical Extraction Report Language (PERL) program and a distributed object, comprising the steps of:

- a) receiving a request from said PERL program, said request specifying said distributed object;
- b) translating said request from said PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA);
- c) making a call to access said distributed object via the Common Object Request Broker Architecture (CORBA);
- d) receiving a response from said call in said step c);
- e) translating said response to a form which is substantially compliant with the Practical Extraction Report Language; and
- f) passing said translated response from said step e) to said PERL program (emphasis added).

Claim 1 recites a limitation of translating a request from a PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA). Applicants respectfully assert that the combination of Mowbray and UCS fails to teach or suggest this claimed limitation, alone or in combination.

Mowbray fails to teach or suggest translating a request from either PERL or a scripting language to CORBA. UCS may assert that PERL is a scripting language. However, UCS fails to teach or suggest translating a request from either PERL or a scripting language to CORBA. Therefore, the combination of Mowbray and UCS fails to teach or suggest the claimed limitation of translating a request from a PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA), as claimed.

Mowbray Fails to Teach or Suggest Translating a Request from either PERL
or a Scripting Language to a Format Suitable for CORBA

Mowbray from pages 35-53 presents an introduction to CORBA. At page 38, Mowbray discloses that vendors have implemented mappings from OMG (Object Management Group) IDL (Interface Definition Language) to programming languages such as C, C++, and Smalltalk. However, Applicants note that Mowbray fails to mention PERL as one of the languages for which a mapping currently exists. Further, Applicants note that the above list of languages Mowbray provides does not include any scripting languages.

Mowbray may suggest at the top of page 39 that potentially there could be a mapping from OMG IDL to other programming languages. However, Applicants do not understand Mowbray to provide any teaching or suggestion as to how such potential mappings could be accomplished. Applicants note

that Mowbray is silent in this passage as to the possibility for mappings between OMG IDL and scripting languages.

Mowbray provides an example of a mapping from OMG IDL to C on pages 39-40. However, Mowbray emphasizes that there will be significant differences for mappings to other languages. Therefore, Mowbray actually suggests the difficulty of providing mappings between OMG IDL and other languages. For example, Applicants note that in the second paragraph of page 40 Mowbray teaches that “[o]ther OMG IDL language mappings appear quite different from the C mapping.” Mowbray goes on to explain that renaming of “type Object” does not seem to be an important factor in C, but the specialization of “type Object” is an important factor in object-oriented languages such as C++ and CLOS. Thus, Mowbray teaches that there are significant differences between the different mappings from OMG IDL and different languages. Applicants do not understand Mowbray to provide any guidance as to how to provide a mapping that can handle such differences. Thus, Applicants respectfully assert that it would not have been obvious to one of ordinary skill in the art at the time of Applicants’ invention to modify the teachings in Mowbray of providing a mapping from OMG IDL to C to arrive at the limitations in Applicants Claim 1.

Mowbray further indicates that for C, there is a complex set of conventions for how parameters are passed. Applicants believe that this is true of most languages, including PERL. However, while Mowbray may provide some teaching as to a “C mapping” at the bottom of page 39, Applicants do not

understand Mowbray to provide any guidance with respect to either a scripting language or PERL. Applicants also note that on the bottom of page 40 Mowbray states that there exist standardized mappings by OMG for C, C++, and Smalltalk. However, Applicants note that Mowbray is silent as to the existence of any standardized mappings for either a scripting language or PERL. Hence, Applicants do not understand these passages to provide guidance as to the claimed limitation of “translating said request from said PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA).” Nor do they provide guidance for translating a request from a scripting language to a format that is suitable for CORBA.

Mowbray at page 253 discusses an example, which Mowbray describes as “wrapping with scripts.” Based on the page 2 of the rejection, Applicants understand the rejection to assert that the combination of the passages on page 253-254 and 35-53 teaches the presently discussed claim limitation. Applicants have already argued that pages 35-53 fail to teach or suggest mapping between OMG IDL and a scripting language. The passage at page 253-254 fails to remedy this deficiency. In fact, Applicants do not understand the passage at page 253-254 to provide any guidance as to providing a mapping between OMG IDL and a scripting language. Rather, the passage at page 253-254 is one in a series of examples that discusses wrapping, as opposed to providing a mapping between OMG IDL and a given language. In particular, the example at pages 253-254 discusses a program “Rapport” that may provide control of its functions via an API, which is referred to by Mowbray as a detailed script API. However, Applicants do not understand Mowbray’s use of

the term “script” in this context to provide any teaching or suggestion of the presently discussed claim limitation, even if combined with the teachings at pages 35-53.

The Combination of Mowbray and UCS Fails to Teach or Suggest Translating
a Request from PERL to a Format Suitable for CORBA

For reasons above, Mowbray fails to teach or suggest, “translating a request from either a scripting language or a PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA).” The combination of Mowbray and UCS fails to teach “translating a request from a PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA),” as claimed. This is because even if UCS were to be combined with Mowbray, the combination would not result in the claimed limitation. The reason for this is that although UCS may assert that PERL is a scripting language, neither UCS nor Mowbray teach or suggest translating a request from either PERL or a scripting language to a format suitable for CORBA.

Claim 1 further recites, “translating said response to a form which is substantially compliant with the Practical Extraction Report Language.” For the foregoing reasons, the combination of Mowbray and UCS fails to teach or suggest this claimed limitation.

For the foregoing rationale, the combination of Mowbray and UCS fails to teach or suggest the limitations of Claim 1. As such, Applicants respectfully request the allowance of Claim 1.

Claim 11 recites, in part:

b) translating said request from said PERL program to a format which is suitable for use with a Common Object Request Broker Architecture (CORBA);

...
e) translating said response to a form which is substantially compliant with the Practical Extraction Report Language.

For the reasons discussed in the response to Claim 1, the combination of Mowbray and UCS fails to teach or suggest the limitations of Claim 11. As such, Applicants respectfully request the allowance of Claim 11.

Claim 20 recites, in part:

a) means for translating a call from said PERL program to a format substantially compliant with a Common Object Request Broker Architecture (CORBA); and

b) means for translating a response from said call to a format substantially compliant with the Practical Extraction Report Language.

For the reasons discussed in the response to Claim 1, the combination of Mowbray and UCS fails to teach or suggest the limitations of Claim 20. As such, Applicants respectfully request the allowance of Claim 20.

Claims 2-10, 12-19, and 21-23 depend from Claims 1, 11, and 20, which are believed to be allowable for the foregoing reasons. By virtue of their dependencies, Claims 2-10, 12-19, and 21-23 are believed to be allowable and Applicants earnestly request their allowance.

CONCLUSION


Based on the arguments presented above, it is respectfully submitted that Claims 1-23 overcome the rejections of record. Therefore, allowance of Claims 1-23 is respectfully solicited.

Should the Examiner have a question regarding the instant response, the Applicants invite the Examiner to contact the Applicants' undersigned representative at the below listed telephone number.

Dated: 12/15, 2003

Respectfully submitted,

WAGNER, MURABITO & HAO LLP


Ronald M. Pomerence
Registration No. 43,009

Address:

WAGNER, MURABITO & HAO LLP
Two North Market Street
Third Floor
San Jose, California 95113

Telephone:

(408) 938-9060 Voice
(408) 938-9069 Facsimile